Amendments to the Claims

Please cancel claims 1 and 7. Please amend claims 2, 4, 8 and 9. The currently pending claims after amendment are listed below.

1. (Cancelled)

2. (Currently Amended) The method for database query optimization of ~~claim 1~~ claim 5, wherein said respective processor resource assignment of each partition comprises a respective number of virtual processors of each partition, said respective number being an integer.

3. (Original) The method for database query optimization of claim 2, wherein said step of defining a plurality of logical partitions comprises defining at least one set of processors which is shared by a set of said logical partitions, said set of said logical partitions containing at least two partitions, said respective processor resource assignment of each partition of said set of partitions including said set of processors.

4. (Currently Amended) The method for database query optimization of ~~claim 1~~ claim 6, further comprising the step of:

saving said first search strategy in a persistent object for later execution, said saving step including saving said first processor resource assignment in said object.

5. (Previously Presented) A method for database query optimization in a computer system having a plurality of central processors, comprising the steps of:

defining a plurality of logical partitions of said computer system, each logical partition having a respective processor resource assignment, wherein each task executing in said computer system is assigned to a respective one of said logical partitions and wherein the definition of a plurality of logical partitions may be dynamically altered;

defining a database query;

constructing a first search strategy for said database query, said first search strategy being dependent on a first processor resource assignment at the time said step of constructing a first search strategy is performed;

invoking said database query for execution in a first logical partition, said invoking step being performed after said step of constructing a first search strategy;

comparing a second processor resource assignment to said first processor resource assignment, said second processor resource assignment being associated with said first logical partition at the time said invoking said database query for execution step is performed;

automatically constructing a second search strategy dependent on said second processor resource assignment, said step of automatically constructing a second search strategy being performed dependent on the results of said comparing step;

saving said first search strategy in a persistent object for later execution, said saving step including saving said first processor resource assignment in said object;

invoking a previously saved search strategy for execution in a second logical partition, said second logical partition being different from said first logical partition;

identifying a third processor resource assignment associated with said second logical partition;

comparing said third processor resource assignment to said first processor resource assignment; and

automatically constructing a third search strategy for execution of said database query depending on the results of said comparing step.

6. (Previously Presented) A method for database query optimization in a computer system having a plurality of central processors, comprising the steps of:

defining a plurality of logical partitions of said computer system, each logical partition having a respective processor resource assignment, wherein each task executing in said computer system is assigned to a respective one of said logical partitions and wherein the definition of a plurality of logical partitions may be dynamically altered;

defining a database query;

constructing a first search strategy for said database query, said first search strategy being dependent on a first processor resource assignment at the time said step of constructing a first search strategy is performed;

invoking said database query for execution in a first logical partition, said invoking step being performed after said step of constructing a first search strategy;

comparing a second processor resource assignment to said first processor resource assignment, said second processor resource assignment being associated with said first logical partition at the time said invoking said database query for execution step is performed;

determining whether a user has specified that automatic construction of another search strategy be disabled;

automatically constructing a second search strategy dependent on said second processor resource assignment, said step of automatically constructing a second search strategy being performed dependent on the results of said comparing step, wherein said step of automatically constructing a second search strategy dependent on said second processor resource assignment is performed only if said determining step determines that a user has not specified that automatic construction of another search strategy be disabled.

7. (Cancelled)

8. (Currently Amended) The program product for database query optimization of ~~claim 7~~ claim 11, wherein said respective processor resource assignment of each partition comprises a respective number of virtual processors of each partition, said respective number being an integer.

9. (Currently Amended) The program product for database query optimization of ~~claim 7~~ claim 11, wherein said instructions further cause said computer system to perform the step of:

saving said first search strategy in a persistent object for later execution, said saving step including saving said first processor resource assignment in said object.

10. (Previously Presented) A program product for database query optimization in a computer system having a plurality of central processors and a dynamic logical partitioning mechanism, said dynamic logical partitioning mechanism supporting a plurality of defined logical partitions of said computer system, each logical partition having a respective processor resource assignment, wherein each task executing in said computer system is assigned to a respective one of said logical partitions and wherein the definition of a plurality of logical partitions may be dynamically altered, said program product comprising a plurality of processor executable instructions recorded on signal-bearing media, wherein said instructions, when executed by at least one central processor of said computer system, cause said computer system to perform the steps of:

receiving a definition of a database query;

constructing a first search strategy for said database query, said first search strategy being dependent on a first processor resource assignment at the time said step of constructing a first search strategy is performed;

invoking said database query for execution in a first logical partition, said invoking step being performed after said step of constructing a first search strategy;

comparing a second processor resource assignment to said first processor resource assignment, said second processor resource assignment being associated with said first logical partition at the time said invoking said database query for execution step is performed;

automatically constructing a second search strategy dependent on said second processor resource assignment, said step of automatically constructing a second search strategy being performed dependent on the results of said comparing step;

saving said first search strategy in a persistent object for later execution, said saving step including saving said first processor resource assignment in said object;

invoking a previously saved search strategy for execution in a second logical partition, said second logical partition being different from said first logical partition;

identifying a third processor resource assignment associated with said second logical partition;

comparing said third processor resource assignment to said first processor resource assignment; and

automatically constructing a third search strategy for execution of said database query depending on the results of said comparing step.

11. (Previously Presented) A program product for database query optimization in a computer system having a plurality of central processors and a dynamic logical partitioning mechanism, said dynamic logical partitioning mechanism supporting a plurality of defined logical partitions of said computer system, each logical partition having a respective processor resource assignment, wherein each task executing in said computer system is assigned to a respective one of said logical partitions and wherein the definition of a plurality of logical partitions may be dynamically altered, said program product comprising a plurality of processor executable instructions recorded on signal-bearing media, wherein said instructions, when executed by at least one central processor of said computer system, cause said computer system to perform the steps of:

receiving a definition of a database query;

constructing a first search strategy for said database query, said first search strategy being dependent on a first processor resource assignment at the time said step of constructing a first search strategy is performed;

invoking said database query for execution in a first logical partition, said invoking step being performed after said step of constructing a first search strategy;

comparing a second processor resource assignment to said first processor resource assignment, said second processor resource assignment being associated with said first logical partition at the time said invoking said database query for execution step is performed;

determining whether a user has specified that automatic construction of another search strategy be disabled;

automatically constructing a second search strategy dependent on said second processor resource assignment, said step of automatically constructing a second search strategy being performed dependent on the results of said comparing step, wherein said step of automatically constructing a second search strategy dependent on said second processor resource assignment is performed only if said determining step determines that a user has not specified that automatic construction of another search strategy be disabled.

12. (Previously Presented) A computer system, comprising:

a plurality of central processing units;

a memory;

a logical partitioning mechanism supporting a plurality of defined logical partitions of said computer system, each logical partition having a respective processor resource assignment, wherein each task executing in said computer system is assigned to a respective one of said logical partitions and wherein the definition of said logical partitions may be dynamically altered;

a database;

a database management system for managing said database, wherein said database management system:

(a) performs query optimization of a database query for said database to produce a first search strategy, said first search strategy being dependent on a first processor resource assignment;

(b) responsive to invoking said first query search strategy for execution, compares said first processor resource assignment with a second processor resource assignment associated with a logical partition of execution at the time said first search strategy is invoked for execution; and

(c) depending on the results of said comparison performed in (b), automatically constructs a second search strategy dependent on said second processor resource assignment;

wherein said database management system further determines whether a user has specified that (c) be disabled, and disables (c) responsive to determining that a user has so specified.

13. (Original) The computer system of claim 12, wherein said respective processor resource assignment of each partition comprises a respective number of virtual processors of each partition, said respective number being an integer.

14. (Original) The computer system of claim 13, wherein said logical partitioning mechanism supports the definition of at least one set of processors which is shared by a set of said logical partitions, said set of said logical partitions containing at least two partitions, said respective processor resource assignment of each partition of said set of partitions including said set of processors.

15. (Original) The computer system of claim 12, wherein said database management system saves said first search strategy in a persistent object for later execution, said persistent object including said first processor resource assignment.

16. (Cancelled)